

Procédure de pérennisation des scripts IGEPP

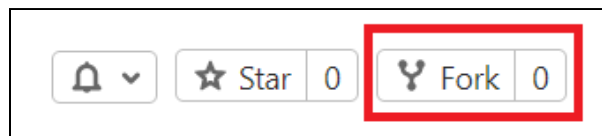
Dépôt sur <https://forgemia.inra.fr>

Dépôt des scripts

Demandez l'accès au répertoire de votre équipe

Vous pouvez contacter gogepp@inrae.fr pour faire la demande

“Forkez*” le projet de votre équipe sur <https://forgemia.inra.fr>




Choisissez de forker sous votre nom d'utilisateur




“Clonez*” votre copie localement

Clone with SSH

`git@forgemia.inra.fr:mateo.boud` 

Clone with HTTPS

`https://forgemia.inra.fr/mateo.` 

Open in your IDE

Visual Studio Code (SSH)

Visual Studio Code (HTTPS)

```
git clone git@forgemia.inra.fr:mateo.boudet/debi.gitgit@forgemia.inra.fr:mateo.boudet/debi.git
```

Créez un dossier à votre nom (si il n'existe pas)

```
$ cd debi/  
/debi$ mkdir mboudet
```

Placez vos scripts dans ce dossier

Si vous le jugez nécessaire, il est possible de créer des sous-dossiers (ex: nom de projet, technologie) pour améliorer la visibilité.

```
cd mboudet/  
mboudet$ mkdir projet_BrasExplor  
mboudet$ cp monscript.py projet_BrasExplor/
```

Quels fichiers sont nécessaires?

- Vos scripts
- Toutes les informations nécessaires pour “Fixer” les versions utilisées
 - `pip freeze` pour les virtualenv
 - `conda list` pour les environnement conda
 - Dans le cas de workflows, spécifier dans les workflows les versions si possible

Quels fichiers sont optionnels (mais recommandés)?

- Si nécessaire, les ressources minimales recommandées (CPU/RAM/Espace disque)
- Un jeu de données d'entrée et de sortie pour tester le script (ou lien vers ces données)

[Ajoutez les fichiers](#) que vous avez créé/modifié au “commit*” actuel

```
/debi/mboudet$ git add projet_BrasExplor/myscript.py
```

[Créez votre commit](#) avec un message adapté

```
git commit -m "Ajout scripts mboudet"
```

[“Pushez*” vos modifications](#)

Vos modifications sont maintenant visibles sur la forgeMia (sur votre fork)

[Intégrez vos modifications](#) au projet de l'équipe

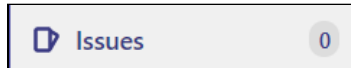
Pour le moment, vos modifications ne sont présentes que sur votre copie du projet. Il vous faut maintenant faire une “merge request” pour intégrer ces modifications au code commun. Il restera ensuite à GOGEP de valider ou non vos modifications.

Ajout de votre protocole au catalogue

Les “Issues*” de la forgeMia vont servir de catalogue de protocoles pour votre équipe.

Créez une nouvelle issue

Sur la page de votre équipe, cliquez sur “Issues” dans la barre de gauche, puis “New issue”.



Les informations

Choisissez un titre adapté (La question à laquelle répond votre protocole par exemple), et incluez dans la description le contenu de votre protocole (les commandes à lancer, les options, etc...).

Il est possible d’inclure un fichier texte à l’issue si vous avez déjà ces informations sous format texte.

Il est également possible d’éditer et d’ajouter d’autres informations après la création de l’issue.

Les labels

Les labels vont servir à transformer cette liste d’issues en catalogue de protocoles. Il est possible de “taguer” chaque issue avec un ou plusieurs labels. Ces labels peuvent être le nom d’un organisme, d’une technologie, ou de n’importe quelle information vous semble pertinente. Sur la page de l’issue, cliquez sur le bouton ‘Edit’ dans la catégorie “Labels” de la barre de droite.



Vous pouvez alors sélectionner un label existant, ou en créer un nouveau si celui que vous cherchez n’existe pas.

Les scripts

Si vous avez intégré vos propres scripts au projet d’équipe, merci d’indiquer dans l’issue un lien direct vers le dossier qui contient ces scripts (ou le script lui-même).

Vous pouvez simplement utiliser l’url qui pointe vers ce dossier/fichier
(ex: <https://forgemia.inra.fr/mateo.boudet/debi/-/blob/master/README.md>)

FAQ

Comment “Cloner” depuis la forgeMia?

Il est possible d’interagir avec la forgeMia de deux façons :

- Via SSH

Pour pouvoir puller en utilisant l'adresse SSH (ex [git@forgemia.inra.fr:igepp/debi.git](ssh://git@forgemia.inra.fr:igepp/debi.git)), il est nécessaire d'avoir une paire de clé SSH (comme pour Genouest). Il vous faudra ajouter votre clé publique dans l'interface de la forge (Profile -> SSH Keys)

- Via HTTP

Pour pouvoir vous authentifier avec l'adresse HTTP, il vous faut créer un token d'accès a l'adresse https://forgemia.inra.fr/-/profile/personal_access_tokens.

Le token créé doit avoir "read_repository" et "write_repository".

Gardez le token généré quelque part!

Ensuite, quand vous ferez un git "clone" ou git "push", il vous faudra spécifier ce token a la place de votre mot de passe.

Dans tous les cas, il vous suffit maintenant de lancer la commande `git clone l'adresse_du_repo`` pour créer votre copie locale.

Comment ajouter un fichier au 'commit' actuel?

Il vous suffit d'utiliser la commande `git add`` avec le chemin de votre fichier

NB: Cette commande doit être lancée depuis l'intérieur du dossier du projet

Git propose automatiquement en auto-complétion les fichiers qui ont été modifiés/ajoutés

Si vous ajoutez un fichier non modifié, il ne se passera rien.

Comment créer un "commit"?

Après avoir ajouté vos fichier, vous pouvez simplement utiliser la commande `git commit -m *Votre message de commit*`

Cette commande agrège tous vos changements et crée un commit avec le message que vous lui donnez.

NB: Vous pouvez créer plusieurs commits avant de les envoyer vers le répertoire distant

Par exemple :

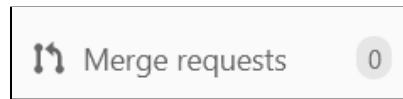
- Je modifie un fichier
- J'ajoute le fichier et je crée le commit 'commit1'
- Je modifie un autre fichier (ou le même)
- J'ajoute le fichier et je crée le commit 'commit2'
- J'envoie les modifications
- Les deux commits apparaissent séparément dans l'interface

Comment 'Pusher' vos modifications?

Après avoir créé vos commits, il vous suffit de faire la commande `git push``. Selon la méthode que vous avez choisie pour cloner le projet distant, il pourra être nécessaire de spécifier à nouveau votre nom d'utilisateur et token.

Comment faire une “merge request”?

Sur la page de la forgeMia de votre équipe (ou celle de votre fork), cliquez sur “Merge request” dans la barre de gauche.



Ensuite, choisissez comme “Source” (point d’origine) votre fork (et la branche *master*), et comme “Target” (cible) le répertoire de votre équipe.

Après avoir cliqué sur “*Compare branches and continue*”, une liste des modifications que vous avez faites est affichée. Vous pouvez changer le nom de la “Merge request”, et ajouter une description. Cliquez ensuite sur “Create merge request”.

Votre “merge request” sera ensuite validée par les membres de GOGEP.

Glossaire

- “Fork”: Copie un projet existant sous votre nom, pour y faire des modifications
- “Cloner” : Rapatrier en local un projet distant
- “Commit”: Un ou plusieurs changements dans des fichiers locaux
- “Committer”: Rassembler les changements faits sous un nom compréhensible (ex: *Modification fichier README*)
- “Pusher”: “Pousser” les commits locaux vers votre projet distant
- “Merger”: Faire une demande pour intégrer vos changements au projet principal
- “Issue”: Une discussion sur un point particulier